EXPLOITING MPEG-7 TEXTURE DESCRIPTORS FOR FAST H.264 MODE DECISION

N. Kamnoonwatana, D. Agrafiotis and C. N. Canagarajah

Department of Electrical & Electronic Engineering, University of Bristol, Bristol, BS8 1UB, UK

ABSTRACT

A novel technique for exploring the use of indexing metadata in improving coding efficiency is proposed in this paper. The technique uses an MPEG-7 descriptor as the basis for a fast mode decision algorithm for H.264/AVC encoders. The descriptor is used to form homogenous clusters for each frame, within which limited available coding modes for each macroblock are defined. The coding mode of an already coded macroblock that belongs to the same cluster in the same frame as well as the statistics of the coding modes of similar clusters in previous frames, are used for limiting the range of available coding modes within each cluster. The results show that the proposed algorithm is able to achieve an average of 47% time-saving when compared to the full search method and 21% when compared to the fast mode decision algorithm employed in the JM12.2 reference H.264 software encoder. In both cases, results yield only a small degradation in rate-distortion performance and a negligible loss in subjective quality.

Index Terms— Video coding, H.264, Mode decision, MPEG-7, Metadata

1. INTRODUCTION

The H.264/AVC video coding standard [1] offers high coding efficiency when compared to prior standards such as MPEG-2/H.262,, H.263, and MPEG-4 Visual. One of the features that contributes to this improvement is the various macroblock (MB) coding modes supported by H.264. These modes can be categorised into intra and inter prediction modes, with the inter prediction modes corresponding to the different types of macroblock partitioning for motion estimation and the intra modes corresponding to the size and type of the prediction. The H.264 standard allows seven partitioning types for inter coded MBs, namely 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4 for the luminance component and half these sizes for the chroma components. Similarly, intra modes correspond to two prediction block sizes of 16x16 and 4x4 pixels for the luma component and one block size (8x8) for the chroma component, with each mode being associated with a number of prediction directions/types.

In order to choose the best mode for each macroblock, and hence achieve the best coding efficiency, a rate distortion optimisation (RDO) technique is recommended for use at the encoder [2]. This technique exhaustively calculates the rate-distortion cost of every mode and then chooses the one that offers the smallest cost. Clearly this is not a computationally effective method and there have been a number of studies on how to improve the coding mode decision process in terms of minimising the complexity (i.e. computational time) whilst keeping any rate-distortion performance penalty as small as possible, including that of [3]. The current JVT software (JM12.2) uses the methods suggested in [4] for its fast mode decision feature namely early-skip mode decision and selective intra mode decision. The two techniques benefit from knowing the behaviour of mode selection in natural video sequences where SKIP mode is most likely to be chosen as the best mode while inter mode is more likely to be chosen than intra mode.

The proposed fast mode decision approach is based on the observation that there are homogeneous regions in most natural video sequences with the macroblocks of each region often undergoing the same motion [3]. Hence macroblocks that belong to the same homogeneous regions are likely to be coded in a similar manner. In order to determine such homogeneous regions the use of an MPEG-7 texture descriptor [5] together with clustering is proposed. Since it is believed that in the near future indexing metadata such as MPEG-7, will be available alongside the video content, the encoder may as well exploit such data for purposes other than just accessibility, such as compression and coding efficiency. The idea of using MPEG7 descriptors for coding purposes has also been explored in [6].

This work is an extension to a previous work of ours [7]. The main contribution is that the new method takes into account the coding mode statistics of each cluster, thus allowing the algorithm to exploit the use of homogeneous clusters spatially and temporally. The paper is organized as follows. Section 2 describes the observations made regarding the coding modes used in different homogeneous regions. Section 3 explains the clustering process of the texture descriptor values. The proposed mode decision algorithm is described in section 4. Section 5 discusses the experimental results and finally section 6 concludes the paper.

2. CODING MODE OBSERVATIONS

As mentioned earlier, the proposed algorithm is based on the observation that macroblocks in homogeneous regions of

video sequences are often coded in similar modes. An example of this observation is given in Fig.1 where the coding modes chosen by the exhaustive mode decision process of H.264 are shown. It can be seen that in the *Flower garden* sequence, most of the macroblocks in the area of the flowers (bottom of the frame) are encoded using inter-prediction with a small partitioning size (i.e. 8x8 and smaller), whereas the macroblocks in the sky region are coded mostly in SKIP mode and large partitioning sizes of inter-prediction modes (i.e. 16x16, 16x8 and 8x16). This behaviour can also be observed in the *Stefan* sequence where MBs in the court area are mostly coded with SKIP mode while the crowd in the background are coded using small partitioning inter-modes. Fig. 2 illustrates how the macroblocks in the sky region of the *Flowergarden* sequence are coded for the first 30 frames. It shows that MBs of the same homogenous region tend to be coded with similar modes over time. Note that *INTER BIG* indicates 16x16, 16x8 and 8x16 inter-prediction modes, and *INTER SMALL* includes 8x8, 8x4, 4x8 and 4x4 inter-prediction modes.

Therefore we can expect that if these homogenous regions are known, the range of possible coding modes can be reduced, by making use of prior information regarding the coding mode of already coded MBs of the same region in the current and previous frame.

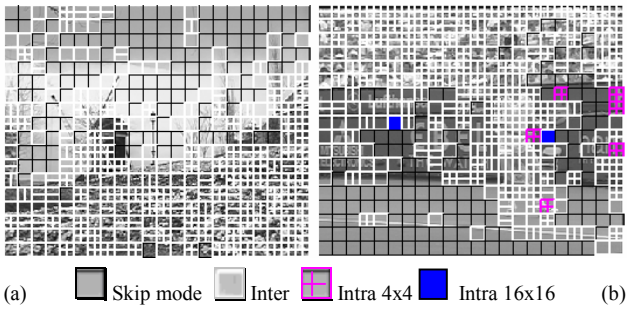

Fig. 1. Coding modes for one frame of (a) *Flowergarden* (b) *Stefan*

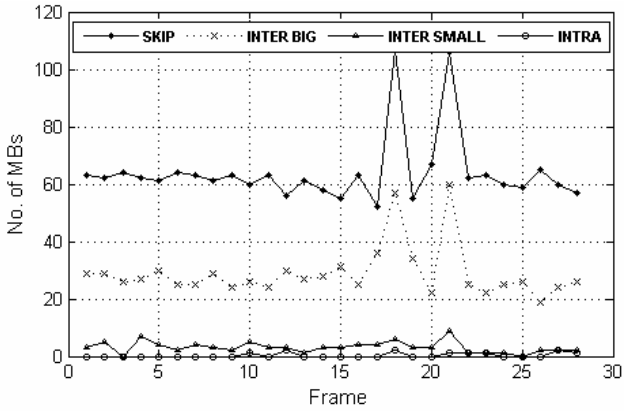

Fig. 2. Coding modes for MBs in the sky region of *Flowergarden*

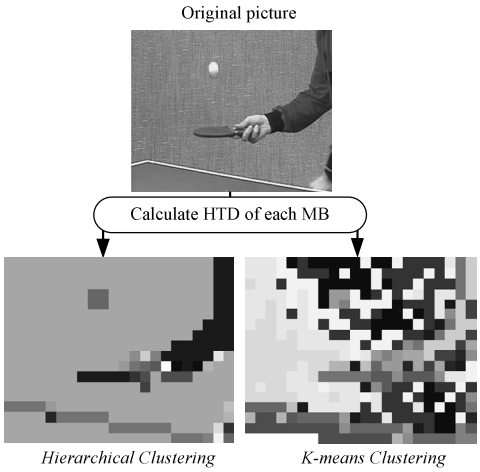

Fig. 3. An example of HTD clustering

3. HOMOGENEOUS REGION CLUSTERING

To form homogeneous clusters a criterion of homogeneity must be used. This criterion can be based on many factors such as intensity, colour, and texture. In this paper the homogeneity criterion used, is based on the MPEG-7 Homogeneous Texture Descriptor (HTD). The descriptor itself is described in [8] as a useful similarity measure and also an effective characterisation of homogeneous texture regions. The HTD has the syntax shown in (1) where f_{DC} is the average intensity of the image, f_{SD} is the standard deviation of the image and e_i and d_i are the energy and energy deviation corresponding to scale-orientation pair i respectively

$$HTD = [f_{DC}, f_{SD}, e_1, e_2 \ldots, e_{30}, d_1, d_2, \ldots, d_{30}] \quad (1)$$

In each frame the MPEG-7 Homogenous Texture Descriptor is calculated for every macroblock, therefore each macroblock in a frame has an HTD vector associated with it. These HTD vectors are then used to form homogeneous texture clusters by a clustering algorithm.

There are many clustering algorithms that can be used to group data based on some similarity decisive factor. An overview of this topic can be found in [9]. This paper does not focus on the clustering algorithm but rather on how to apply the result of the clustering to the mode decision process of H.264. An example of clustering of HTD values is shown in Fig. 3 where the results obtained from a k-means and a hierarchical clustering algorithm are shown. Note that for both algorithms the number of clusters was limited to 20. The clusters are illustrated in greyscale such that macroblocks with the same colour belong to the same cluster. It is clear that the hierarchical algorithm offers a better result in terms of its ability to distinguish different types of texture and to group similar textures based on their HTD values. The hierarchical algorithm was selected in the implementation of the proposed algorithm.

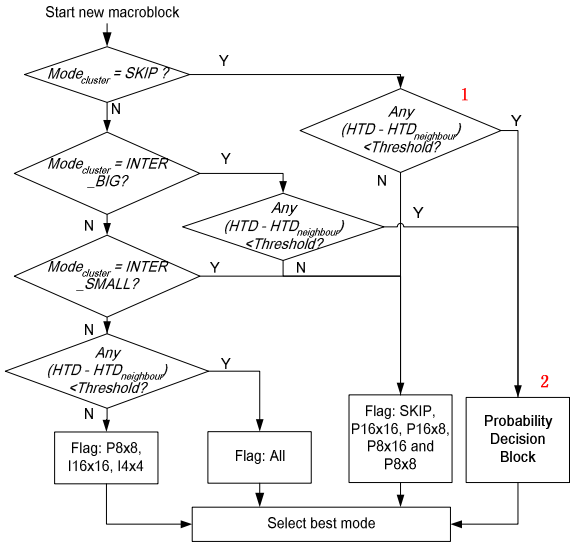

Fig. 4. Flowchart of the proposed algorithm

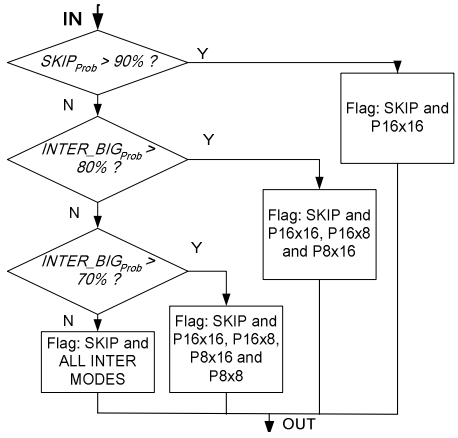

Fig. 5. Flowchart of the Probability Decision Block

4. PROPOSED FAST MODE DECISION

The aim of the proposed algorithm is to narrow down the choices of coding modes for each macroblock based on the information coming from other coded macroblocks that belong to the same homogeneous cluster. The scheme of the proposed algorithm is shown in Fig. 4. Note that the algorithm is activated in P-slices and not on the first macroblock of each cluster.

The algorithm first checks what cluster the current macroblock belongs to. The algorithm then checks the mode of the previously coded macroblock of the same cluster. This particular mode is referred to as $Mode_{cluster}$ in the diagram where *INTER_BIG* indicates 16x16, 16x8 and 8x16 inter-prediction modes, and *INTER_SMALL* indicates 8x8, 8x4,4x8 and 4x4 inter-prediction modes. The algorithm decides which modes will be included in the rate-distortion cost calculation. Note that there are two additional decision blocks, as illustrated in Fig. 4. Decision block *1* indicates that the algorithm takes into account the difference between the HTD value of the current block (*HTD*) and the HTD value of the temporally neighbouring blocks (HTD_{neigh}) in the reference frame. The range of these neighbouring blocks matches the search range selected for the motion estimation in the encoder. This decision block represents a texture similarity criterion that checks if there is any block within the neighbourhood of the reference frame with a similar homogeneous texture. The main purpose of this decision block is to improve the probability of selecting the right range of possible coding modes. Decision block *2* takes into account the probabilities of coding mode occurrence for the same cluster from the previous frame. This block can be broken down into smaller decision blocks as shown in Fig. 5. The purpose of decision block *2* is to further narrow down the available choices of coding mode based on the coding modes selected in the previous frame.

The proposed algorithm can be integrated with the existing fast mode decision methods used in JM12.2 namely *Early SKIP detection* and *Selective Intra mode decision*. It can be placed right after the early SKIP mode testing. If the conditions for the SKIP mode are met then this algorithm will not be executed, on the other hand, if these conditions are not met then the algorithm will execute as usual.

5. EXPERIMENTAL RESULTS

The proposed fast mode decision algorithm was implemented in the JM12.2 reference H.264 software and all experimental results were carried out on an Intel Pentium 4 machine with a 2.26GHz CPU and 1GB of RAM. The following parameters were used for obtaining the results. The MV search range was set to 16 with 1 reference frame and a GOP structure of IPPP…I, with one I frame every 30 frames. The motion estimation scheme was set to fast full search. The first 100 frames of a number of test sequences were coded with four quantisation parameters, namely 25, 30, 35, and 40. The hierarchical clustering method was used with a Euclidean distance metric and with the number of clusters set to 40. Note that the number of clusters can be varied, thus offering a mechanism for controlling the trade-off between complexity and rate-distortion performance. This will be demonstrated later in this section.

The performance results presented are given as bit rate, PSNR, and encoding time differences (*ΔRate*, *ΔY-PSNR*, and *ΔTime* respectively) relative to the compared reference method. The method for calculating the bit rate and PSNR differences follows the recommendations of [10]. The calculation of the encoding time difference is done according to equation (2) where $Time_{proposed}$ is the encoding time achieved by the proposed algorithm and $Time_{reference}$ denotes the encoding time of the reference method.

$$\Delta Time(\%) = \left(\frac{Time_{proposed} - Time_{reference}}{Time_{reference}} \right) \times 100 \qquad (2)$$

Table 1 shows the simulation results where the proposed algorithm is compared to the exhaustive mode decision algorithm. The proposed method achieves a 47.48% reduction in encoding time on average, with an increase of 2.58% in bit rate and a penalty of 0.11dB in PSNR. It can be observed that the proposed algorithm works very well in sequences where there are well defined homogeneous texture regions with macroblocks that undergo similar motion; for example, the sequences *Container* and *Flowergarden* that include such regions - the sea and the sky in *Container,* the flowers and sky in *Flowergarden*. In a sequence that contains high and/or complex motion smaller reductions in coding time can be achieved (e.g. *Stefan* and *Calendar*).

Table 2 shows the results of a comparison between the proposed algorithm when integrated with the existing fast mode decision methods in JM12.2, relative to the existing fast mode decision alone. It can be observed that the proposed algorithm is able to achieve an additional 21.39% reduction in encoding time on average over the fast mode decision mechanism of JM12.2 with a negligible increase in bit rate of 1.97% and a reduction in PSNR of 0.08 dB.

The experimental results also suggested that a larger number of clusters can lead to a smaller reduction in coding time with a lower penalty in rate-distortion performance. For example, considering comparison with the exhaustive mode decision (Table 1), when the number of clusters is increased to 80 an average of 36.82% decrease in coding time is achieved with the average increase in bit rate reduced to 1.8% and the PSNR penalty reduced to 0.07 dB.

Table 1. Comparison with exhaustive mode decision

Sequence	**ΔRate(%)**	**ΔY-PSNR(dB)**	**ΔTime(%)**
Calendar	1.118	-0.051	-46.5042
Silent	5.616	-0.219	-49.5397
Stefan	3.501	-0.161	-44.3363
Container	0.843	-0.029	-47.8207
Table tennis	3.019	-0.104	-46.3773
Flowergarden	1.388	-0.076	-50.2867
Average	**2.58**	**-0.11**	**-47.48**

Table 2. Comparison with the JM12.2 fast mode decision

Sequence	**ΔRate(%)**	**ΔY-PSNR(dB)**	**ΔTime(%)**
Calendar	1.049	-0.05	-33.5376
Silent	4.594	-0.189	-14.5673
Stefan	2.383	-0.11	-20.5754
Container	0.605	-0.021	-9.4811
Table tennis	2.027	-0.071	-19.1694
Flowergarden	1.174	-0.066	-31.0334
Average	**1.97**	**-0.08**	**-21.39**

6. CONCLUSIONS

We have presented a fast mode decision algorithm that successfully exploits MPEG-7 metadata in the context of H.264 video coding. The MPEG-7 Homogenous Texture descriptor is used to form homogeneous clusters in every frame. The algorithm exploits the coding mode decision similarities that can be observed in these clusters (both spatially and temporally) so that the possible encoding modes for each macroblock are limited. As a result the proposed method reduces the encoding time significantly without introducing any noticeable rate-distortion penalty.

REFERENCES

[1] Iain E.G. Richardson, H.264 and MPEG-4 Video Compression – Video Coding for Next-generation Multimedia, John Wiley & Sons, England, 2003.

[2] Gary Sullivan, Thomas Wiegand, Keng-Pang Lim, "Joint Model Reference Encoding Methods and Decoding Concealment Methods", JVT-I049, San Diego, USA, September, 2003.

[3] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rachardja, C. C. Ko, "Fast Intermode Decision in H.264/AVC Video Coding", IEEE Trans. Circuits Syst. Video Technol., Vol. 15, No. 6, pp. 953-958, July 2005

[4] B. Jeon, J. Lee, "Fast mode decision for H.264", presented at the 10th JVT-J033 Meeting, Antalya, Turkey, Dec. 2003.

[5] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, A. Yamada, "Color and Texture Descriptors", IEEE Trans. Circuits Syst. Video Technol., Vol. 11, No. 6, pp. 703-715, July 2001.

[6] Javier R. Hidalgo, Philippe Salembier, "On the Use of Idexing Metadata to Improve the Efficiency of Video Compression", IEEE Trans. Circuits Syst. Video Technol., Vol. 16, No. 3, pp. 410 – 419, March 2006.

[7] N. Kamnoonwatana, D. Agrafiotis, N. Canagarajah, "Fast Mode Decision For H.264/AVC Based On Clustering of MPEG-7 Texture Descriptor", PCS, Lisbon, Nov 2007.

[8] Y. M. Ro, M. Kim, H. K. Kang, B.S. Manjunath, and J. Kim, "MPEG-7 Homogeneous Texture Descriptor", ETRI Journal, Vol. 23, No. 2 , pp. 41 -51, June 2001.

[9] A.K. Jain, M.N. Murty, P.J. Flynn, "Data Clustering: A Review", ACM Computing Surveys, Vol. 31, No. 3, pp. 265-323, September 1999.

[10] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M033, 13th VCEG meeting: Austin Texas, USA, April 2001.